#2

Docket No. 1341.1075 (JDH)



**IN THE UNITED STATES PATENT AND TRADEMARK OFFICE**

| | | |
|---|---|---|
| In re Patent Application of: | ) | |
| | ) | |
| Takeshi SHIMOYAMA | ) | Group Art Unit: Unassigned |
| | ) | |
| Serial No.: To be assigned | ) | Examiner: Unassigned |
| | ) | |
| Filed: December 19, 2000 | ) | |

For: **A METHOD AND APPARATUS FOR DESIGNING CIPHER LOGIC, AND A COMPUTER PRODUCT**

**SUBMISSION OF CERTIFIED COPY OF PRIOR FOREIGN APPLICATION IN ACCORDANCE WITH THE REQUIREMENTS OF 37 C.F.R. §1.55**

*Assistant Commissioner for Patents*
*Washington, D.C. 20231*

*Sir:*

In accordance with the provisions of 37 C.F.R. §1.55, the applicant submits herewith a certified copy of the following foreign application:

Japanese Patent Application No. 2000-016413
Filed: January 26, 2000

It is respectfully requested that the applicant be given the benefit of the foreign filing date as evidenced by the certified papers attached hereto, in accordance with the requirements of 35 U.S.C. §119.

Respectfully submitted,

STAAS & HALSEY LLP

By: _____
James D. Halsey, Jr.
Registration No. 22,729

700 11th Street, N.W., Ste. 500
Washington, D.C. 20001
(202) 434-1500
Date: December 19, 2000

PFJA-00043-US

JC960 U.S. PTO
09/739219
12/19/00

# 日 本 国 特 許 庁
PATENT OFFICE
JAPANESE GOVERNMENT

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office.

出 願 年 月 日
Date of Application: ２０００年 １月２６日

出 願 番 号
Application Number: 特願２０００－０１６４１３

出 願 人
Applicant (s): 富士通株式会社

CERTIFIED COPY OF
PRIORITY DOCUMENT

２０００年 ９月１８日

特 許 庁 長 官
Commissioner,
Patent Office 及 川 耕 造

出証番号 出証特２０００－３０７３９５３

【書類名】　特許願
【整理番号】　0050031
【提出日】　平成12年 1月26日
【あて先】　特許庁長官　殿
【国際特許分類】　G06F 15/00
【発明の名称】　暗号設計装置および記録媒体
【請求項の数】　5
【発明者】
【住所又は居所】　神奈川県川崎市中原区上小田中４丁目１番１号　富士通株式会社内
【氏名】　下山　武司
【特許出願人】
【識別番号】　000005223
【氏名又は名称】　富士通株式会社
【代理人】
【識別番号】　100089141
【住所又は居所】　東京都目黒区平町１丁目２１番２０－６０３号
【弁理士】
【氏名又は名称】　岡田　守弘
【電話番号】　03-3725-2215
【手数料の表示】
【予納台帳番号】　015543
【納付金額】　21,000円
【提出物件の目録】
【物件名】　明細書　1
【物件名】　図面　1
【物件名】　要約書　1
【包括委任状番号】　9705795
【プルーフの要否】　要

【書類名】　明細書

【発明の名称】　暗号設計装置および記録媒体

【特許請求の範囲】

【請求項１】

計算機に最適な入出力数のＳ－ｂｏｘを持つ暗号化装置を設計する暗号設計装置において、

計算機の一次キャッシュメモリ量、および全体の入出力ビット数を入力する手段と、

上記入力された全体の入出力ビット数を分割してＳ－ｂｏｘの入出力数を生成し、余りがでたときに任意のＳ－ｂｏｘの入出力数に割り当ててＳ－ｂｏｘの入出力数を仮決定する手段と、

上記仮決定したＳ－ｂｏｘの入出力数を組み合わせて、上記入力された一次キャッシュサイズを越えない範囲で当該組み合わせを行う手段と

を備えたことを特徴とする暗号設計装置。

【請求項２】

上記分割して算出したＳ－ｂｏｘの入出力数の最小値を予め指定したことを特徴とする請求項１記載の暗号設計装置。

【請求項３】

上記全体の入出力ビット数および上記一次キャッシュサイズできまる終了値をもとに、上記組み合わせを終了させることを特徴とする請求項１記載の暗号設計装置。

【請求項４】

上記余りがでたときに可及的に離れた位置のＳ－ｂｏｘの入出力数に割り当てることを特徴とする請求項１記載の暗号設計装置。

【請求項５】

計算機の一次キャッシュメモリ量、および全体の入出力ビット数を入力する手段と、

上記入力された全体の入出力ビット数を分割してＳ－ｂｏｘの入出力数を算出し、余りがでたときに任意のＳ－ｂｏｘの入出力数に割り当ててＳ－ｂｏｘの入

出力数を仮決定する手段と、

上記仮決定したＳ－ｂｏｘの入出力数を組み合わせて、上記入力された一次キャッシュサイズを越えない範囲で当該組み合わせを行う手段と

して機能させるプログラムを記録したコンピュータ読取可能な記録媒体。

【発明の詳細な説明】

【０００１】

【発明の属する技術分野】

本発明は、計算機に最適な入出力数のＳ－ｂｏｘを持つ暗号化装置を設計する暗号設計装置および記録媒体に関するものである。

【０００２】

【従来の技術】

高度情報化社会において高速かつ安全な通信を実現するためには共通鍵ブロック暗号が欠かせない道具であり、通常ユーザが意識することなく、日常的に用いられている。一口に共通鍵暗号と言っても、さまざまな共通鍵暗号アルゴリズムが提案されている。その共通鍵ブロック暗号の多くは、Feistel構造とよばれる単純な繰り返し構造であり、更にその内部構造であるＦ関数とよばれる部分では、通常Ｓ－ｂｏｘと言われる入力ビット数の少ない非線形関数を並べて用い、更にその出力を組み合わせたものを線形関数を用いて拡散させるという方法が最も主流として用いられている。このＦ関数の内部構造は、一般にＳＰＮ(Substitution Permutation Network)構造と呼ばれている。暗号の安全性の核となるのはＳ－ｂｏｘであるため、Ｓ－ｂｏｘの設計は簡単ではない。また、多くの種類のＳ－ｂｏｘを用いる場合にはそれだけ多くの記憶容量が必要となる。そのため開発コスト、必要メモリ量を抑え、あるいは構造を見やすくするために、これまでのＳ－ｂｏｘとしては全く同じものを繰り返し用いるか、あるいは入力サイズ、および出力サイズがそれぞれ等しいＳ－ｂｏｘを使い回して使うことが多かった。

【０００３】

【発明が解決しようとする課題】

共通鍵暗号の入力ビット数は通常６４ビット、あるいは１２８ビットであるた

め、重複なく均等なサイズのＳ－ｂｏｘを用いる場合には４ビット入力、あるいは８ビット入力といった$2^n$ビット入力のＳ－ｂｏｘしか考えられなかった。

【０００４】

Ｓ－ｂｏｘは暗号装置において、最も頻繁に参照されるため、暗号化速度に最も影響を与えやすく、そのためにＳ－ｂｏｘを表す表全体が、計算機の持つ最も高速に参照可能な記憶装置（通常は１次キャッシュメモリ）内に収まるように設計することが望まれる。一方、Ｓ－ｂｏｘの入力ビット数に応じてそのテーブルサイズは指数関数的に大きくなるため現実的に用いられることができるテーブルの大きさには上限がある。もし、記憶装置の容量を越えたテーブルを参照しなければならない場合、アクセス速度は数倍以上遅くなってしまう。暗号設計においては、こうした理由から実装上不利な状況を避けるために、$2^n$入力のＳ－ｂｏｘの選択肢は、現実的には４ビットかあるいは８ビット入力しかとりえなかった。

【０００５】

一方、最近の計算機が持つメモリ容量も年々増加する傾向にある。これらの状況から１６ビット入力のＳ－ｂｏｘを利用するのは、まだ時期尚早であるものの、８ビット入力のＳ－ｂｏｘでは計算機のメモリ資源を十分に生かしているとは言えないという問題もあった。

【０００６】

即ち、小さな入力ビット数を持つＳ－ｂｏｘでは、ほぼ全ての計算機に対して、高速アクセス可能なメモリ内に収めることが可能であるが、分割したＳ－ｂｏｘの総数が増えることが避けられず、メモリ内に記憶されたテーブルを参照する回数が増え、実行速度が低下するという問題があった。

【０００７】

また、大きな入力ビット数を持つＳ－ｂｏｘを使えば、Ｓ－ｂｏｘの総数を減らすことができ、テーブル参照の回数を減らすことができるが、テーブル全体の大きさが大きくなるために、記憶装置に保持できないか、あるいは参照速度が低速な記憶装置にしか保持することができない。従って、１回のテーブル参照にかかる時間そのものが低下し、結果として全体の実行速度が低下するという問題が

あった。

【０００８】

本発明は、これらの問題を解決するため、計算機毎の一次キャッシュメモリ量および全体の入出力ビット数をもとにＳ－ｂｏｘの入出力数を最適化し、計算機毎に最適な高速な暗号化／復号化を実現することを目的としている。

【０００９】

【課題を解決するための手段】

図１を参照して課題を解決するための手段を説明する。

図１において、入力手段２は、パラメータ（計算機の一次キャッシュメモリ量、全体の入出力数、Ｓ－ｂｏｘの最小の入出力数など）を入力するものである。

【００１０】

Ｓ－ｂｏｘの最適化手段３は、Ｓ－ｂｏｘの入出力数の最適化を行うものである。

次に、動作を説明する。

【００１１】

入力手段２が計算機の一次キャッシュメモリ量、および全体の入出力ビット数を入力し、Ｓ－ｂｏｘの最適化手段３が入力された全体の入出力ビット数を分割してＳ－ｂｏｘの入出力数を生成し、余りがでたときに任意のＳ－ｂｏｘの入出力数に割り当ててＳ－ｂｏｘの入出力数を仮決定し、仮決定したＳ－ｂｏｘの入出力数を組み合わせて、入力された一次キャッシュサイズを越えない範囲で当該組み合わせを行うようにしている。

【００１２】

この際、分割して算出したＳ－ｂｏｘの入出力数の最小値を予め指定するようにしている。

また、全体の入出力ビット数および一次キャッシュサイズできまる終了値をもとに、組み合わせを終了させるようにしている。

【００１３】

また、余りがでたときに可及的に離れた位置のＳ－ｂｏｘの入出力数に割り当てるようにしている。

従って、計算機毎の一次キャッシュメモリ量および全体の入出力ビット数をもとにＳ－ｂｏｘの入出力数を最適化することにより、計算機毎に最適な高速な暗号化／復号化を行う装置を実現することが可能となる。

【００１４】

【発明の実施の形態】

次に、図１から図１１を用いて本発明の実施の形態および動作を順次詳細に説明する。

【００１５】

図１は、本発明のシステム構成図を示す。

図１において、処理装置１は、図示外の記録媒体から読み出したプログラムを主記憶にローディングして起動し、以下に説明する各種処理を行うものであって、入力手段２、Ｓ－ｂｏｘの最適化手段３、Ｓ－ｂｏｘの生成手段４、Ｆ関数の生成手段５などから構成されるものである。

【００１６】

入力手段２は、パラメータ（計算機の一次キャッシュメモリ量、全体の入出力数、Ｓ－ｂｏｘの最小の入出力数など）などを入力するものである。

Ｓ－ｂｏｘの最適化手段３は、Ｓ－ｂｏｘの入出力数の最適化を行うものである（図２、図３、図７などを用いて後述）。

【００１７】

Ｓ－ｂｏｘの生成手段４は、最適化されたＳ－ｂｏｘの入出力数に従ったＳ－ｂｏｘを生成するものである（図２、図４、図７から図９などを用いて後述する）。

【００１８】

Ｆ関数の生成手段５は、Ｆ関数を生成するものである（図２、図５、図７、図１１などを用いて後述する）。

入力ファイル６は、計算機の一次キャッシュメモリ量などを格納した計算機パラメータファイル、Ｓ－ｂｏｘに関する各種データを格納したＳ－ｂｏｘのパラメータファイルなどである。

【００１９】

出力ファイル７は、Ｓ－ｂｏｘ（設計／実装）の出力ファイル、Ｆ関数（設計／実装）の出力ファイル、フェイステル構造データ（設計／実装）の出力ファイルなどである。

【００２０】

表示装置８は、画面などを表示するものである。

入出力装置９は、各種入出力装置であって、プリンタ、ディスク装置などである。

【００２１】

次に、図２のフローチャートの順番に従い図１の構成の動作を説明する。

図２は、本発明の全体動作説明フローチャートを示す。

図２において、Ｓ１は、計算機のパラメータの入力を行う。ここでは、計算機のパラメータとして図示の下記を入力する。

【００２２】

・一次キャッシュメモリ量：

①ＰｅｎｔｉｕｍＩＩ：１６キロバイト

②ＰＡ－ＲＩＳＣ：１メガバイト

Ｓ２は、Ｓ－ｂｏｘおよび入出力全体のパラメータを入力する。例えば図示の下記を入力する。

【００２３】

・Ｓ－ｂｏｘの入出力数：５ビット以上

・全体の入出力数：３２ビット

Ｓ３は、Ｓ－ｂｏｘの最適化を行う。これは、第１に、Ｓ２で指定されたＳ－ｂｏｘの入出力数の最小値例えば５ビットで全体の入出力数３２ビットを分割して６個の５ビット

５　５　５　５　５　５

と、余り２ビットとし、この余り２ビットを可及的に離れた位置、例えば左端と右端に割り当てて、

６　５　５　５　５　６

とする。

【００２４】

第２に、２つづつを組にして

１１　１０　１１

と３つに合成し、合成した後の組み合わせ数（ここでは３つ）が、後述する図３で説明するａ＝（（入出力ビット数）／（ｌｏｇ$_2$（キャッシュサイズ））の整数部分）＋１＝（（３２）／（ｌｏｇ$_2$（１６０００））の整数部分）＋１＝３となり、等しいので合成の最適化を終了する（Ｓ１の①の一次キャッシュメモリ量が１６キロバイトの場合）。一方、Ｓ１の②の一次キャッシュメモリ量が１メガバイトの場合には、ａ＝２となり、Ｓ３の右側の③に示すように、３つづつを組にして合成し、

１６　１６

の２組とし、ａ＝２と等しいので最適化を終了する（Ｓ１の②の一次キャッシュメモリが１メガバイトの場合）。

【００２５】

Ｓ４は、Ｓ－ｂｏｘの生成を行う。これは、Ｓ３で最適化した後のＳ－ｂｏｘの入出力数をもとにＳ－ｂｏｘを生成する（図４を用いて後述する）。

Ｓ５は、Ｆ関数の生成を行う。これは、Ｓ４で生成したＳ－ｂｏｘをもとに後述する図５で説明するようにしてＦ関数（Ｆ関数に関する秘密鍵、Ｆ関数入出力ビット数と等しいビット数を持つ鍵など）を作成する。

【００２６】

Ｓ６は、暗号の生成を行う。

以上によって、計算機などのパラメータ（一次キャッシュメモリ量、全体の入出力ビット数）を入力すると、自動的にＳ－ｂｏｘの入出力ビット数の最適化を行い、Ｓ－ｂｏｘの生成、Ｆ関数の生成、および暗号の生成を行うことが可能となる。

【００２７】

図３は、本発明の最適化手順フローチャートを示す。

図３において、Ｓ１１は、５ビットを３２を越えないだけ並べる。これは、既述した図２のＳ２で入力したパラメータ（Ｓ－ｂｏｘの入出力数５ビット以上）

から最小値を５ビットとし、当該Ｓ－ｂｏｘの入出力数の最小値５ビットで全体の入出力数３２ビットを分割して当該３２ビットを越えない分だけ並べる。ここでは、５ビットを６つ下記のように並べる（余りは２である）。

【００２８】

５　５　５　５　５　５

Ｓ１２は、余りを任意の場所に割り当てる。これは、Ｓ１１で並べた余り（ここでは、２）を任意の場所、ここでは、できるだけ離れた場所である、左端と右端にそれぞれ１を割り当てて下記のようにする。

【００２９】

６　５　５　５　５　６

Ｓ１３は、キャッシュサイズを越えない範囲の組み合わせを求める。これは、Ｓ１２で並べたＳ－ｂｏｘの各入出力数を例えば左端から２つづつ組（あるいは３つづつ組）に、下記のようにする。

【００３０】

１１　１０　１１（２つづつ組の場合）

１６　１６　　（３つづつ組の場合）

Ｓ１４は、ａ＝（（全体の入出力ビット数）／（$\log_2$(キャッシュサイズ)）の整数部分）＋１を求める。例えば

ａ＝（３２）／（$\log_2$（１６０００））の整数部分＋１

＝（３２／１２）の整数部分＋１

＝（２．６の整数部分）＋１

＝３

として求める。

【００３１】

Ｓ１５は、Ｓ１３で組み合わせた後の組み合わせ数ｂとＳ１４で求めたａ（組み合わせ数の終了値）とを比較する。ｂ＝ａの場合（Ｓ１３の組み合わせ数ｂが終了値ａ（例えば３）に等しくなった場合）にはＳ１６に進み、最適化終了する。一方、ｂ＞ａの場合には、Ｓ１１に戻り組み合わせを繰り返す。

【００３２】

以上によって、パラメータで指定された最小のＳ－ｂｏｘの入出力数（例えば５ビット）で全体の入出力数（例えば３２ビット）を分割して並べ、余りがでたときは可及的に離れた位置に割り当てて仮のＳ－ｂｏｘの入出力数の組み合わせを作成し、組み合わせ数ｂが全体の入出力ビット数およびキャッシュサイズから求めた終値ａに等しくなるまで組み合わせを繰り返し、最適化を行う。これにより、一次キャッシュメモリに収まる範囲内でＳ－ｂｏｘの組み合わせ数を最小にして参照回数を削減し、計算機毎に個別に最適化を実現することが可能となる。

【００３３】

図４は、本発明のＳ－ｂｏｘの生成フローチャートを示す。

図４において、Ｓ２１は、最適化後の割りあて数を抽出する。例えば図２のＳ３の①の最適化の場合には、６，５を抽出する。

【００３４】

Ｓ２２は、各々の割りあて数に対応する入出力ビット数を持つ非線形型テーブルを作成する。例えば右側に示すように、入力数５ビットをアドレスとし、出力が５ビットの非線形のテーブルを作成する。同様に、６ビットなどの非線形テーブルを作成する。

【００３５】

以上によって、最適化後のＳ－ｂｏｘの組み合わせ数に対応する非線形テーブルを作成できたこととなる。

図５は、本発明のＦ関数例（設計時）を示す。Ｆ関数として、図示のように、図４で作成したＳ－ｂｏｘ（非線形テーブル）を図中のＳのように並べて接続し、上部の全体の入出力数（例えば３２ビット）にＸＯＲ（排他論理和）回路を入れて鍵（例えば３２ビット）との排他論理和演算を行った後の３２ビットをそれぞれ各Ｓ－ｂｏｘに図示のように分割して接続する。各Ｓ－ｂｏｘからの出力ビットをまとめてここでは３２ビットにし、更に、Ｌ（線形変換回路）を通して３２ビットを出力する。

【００３６】

以上の構成により、計算機毎に最適化した入出力数を持つＳ－ｂｏｘを用いたＦ関数を生成（設計）することが可能となる。

図６は、本発明のＦｅｉｓｔｅｌ構造データ例を示す。これは、既述した図５で生成（設計）したＦ関数を用いて生成したＦｅｉｓｔｅｌ構造データをイメージ的に示す。上部から平文（あるいは暗号文）が入力され、矢印のように処理結果が順に流れて下方から暗号文（あるいは平文）が出力されるものであって、暗号化あるいは復号化する回路として動作するものである。この際、図示の構造中のＦ関数を構成するＳ－ｂｏｘの参照が、各計算機毎の一次キャッシュメモリ上でアクセスできるように最適化されているので、当該計算機毎に固有の一次キャッシュメモリ量を最大限有効に活用して高速に暗号化あるいは復号化を実行することが可能となる。

【００３７】

次に、図７から図１１を用いて計算機に実装時に動的に最適化を行うときの動作を順次詳細に説明する。

図７は、本発明の動作説明フローチャート（実装時）を示す。

【００３８】

図７において、Ｓ２１は、計算機のパラメータの入力を行う。これは、本願発明が実装された計算機のパラメータとして下記を入力する（読み込む）。

・一次キャッシュメモリ量：

ＰｅｎｔｉｕｍＩＩ：１６キロバイト

Ｓ２２は、Ｓ－ｂｏｘの最適組の抽出を行う。これは、既述した設計時と同様に、例えば初回は、Ｓ－ｂｏｘの入出力数の最小値である例えば５ビットで全体の入出力数例えば３２ビットを分割して並べ、余りがある場合には可及的に離れた位置に割り当てて

６　５　５　５　５　６

のＳ－ｂｏｘの入出力数の組み合わせを作る。２回目以降は、既述した図３の最適化手順フローチャートに従い２つづつ、あるいは３つづつを組み合わせたりなどして最適な組み合わせを作り、最適化されるまで繰り返す。

【００３９】

Ｓ３３は、Ｓ－ｂｏｘの合成テーブルを作成する。これは、右側に示すように、Ｓ３２で抽出したＳ－ｂｏｘの入出力数の合成テーブル、例えば初回は６，５

のＳ－ｂｏｘの合成テーブルをそれぞれ作成し、２回目な６と５を組にした１１ビットの合成テーブル（拡大Ｓ－ｂｏｘ）などをそれぞれ作成することを繰り返す。

【００４０】

Ｓ３４は、終わりか判別する。ＹＥＳの場合には、Ｓ３５に進む。ＮＯの場合には、Ｓ３２に戻り、次の最適な組の抽出を行うなどを繰り返す。

Ｓ３５は、各々拡大されたＳ－ｂｏｘとＬ（線形変換回路）とを合成する。これにより、Ｌ（線形変換回路）が各々の拡大Ｓ－ｂｏｘに取り込まれ、当該Ｌ（線形変換回路）の処理が不要となり、高速化を図ることが可能となる。

【００４１】

Ｓ３６は、残りの部分を実装する。例えば右側に記載したように、鍵加算部、入出力部などの残りの部分を実装する。

Ｓ３７は、Ｆ関数の実装が終了したこととなる。

【００４２】

以上によって、計算機への実装時に当該計算機からパラメータ（一次キャッシュメモリ量など）を取り込み、この一次キャッシュメモリ量と全体の入出力数をもとに最適なＳ－ｂｏｘの入出力数を求めてＦ関数を自動生成することが可能となる。これにより、当該Ｆ関数を組み込んだ既述した図６のＦｅｉｓｔｅｌ構造データを作成し、平文／暗号文を暗号化／復号化するときに当該計算機が持つ一次キャッシュメモリ上に配置したテーブルを必要最小限の参照回数にして高速化を図ることが可能となる。

【００４３】

図８は、本発明のＳ－ｂｏｘの抽出フローチャートを示す。

図８において、Ｓ４１は、最適化されたＳ－ｂｏｘの組を抽出する。例えば右側に示す下記のように最適化されたＳ－ｂｏｘの入出力数の組を抽出する。

【００４４】

６　５　５　５　５　６

Ｓ４２は、キャッシュメモリ量に応じて組み合わせを抽出する。これは、Ｓ４１で抽出したＳ－ｂｏｘの入出力数の組を例えば左端から２つづつ、あるいは３

つづつ組み合わせて新たな組を作成し、既述した図３のＳ１４のａの終値と等しい組み合わせ数となるまで繰り返し、最適な組み合わせ数を持つＳ－ｂｏｘの入出力数を決定する。例えば図示の下記のように決定（抽出）する。

【００４５】

１１　１０　１１

以上によって、計算機への実装時に当該計算機の一次キャッシュメモリ量に対応した最適なＳ－ｂｏｘの入出力数を決定することが可能となる。

【００４６】

図９は、本発明の合成テーブルの作成フローチャートを示す。

図９において、Ｓ５１は、Ｓ－ｂｏｘの組み合わせを入力する。これは、右側に記載したように、例えばＳ－ｂｏｘの入出力数として６，５を入力する。

【００４７】

Ｓ５２は、Ｓ－ｂｏｘの合成を行う。これは、Ｓ５１で入力されたＳ－ｂｏｘの合成を行い、右側に記載したように、既述した拡大Ｓ－ｂｏｘを作成する。ここでは、例えば右側に示すように、入力数１１ビットをアドレスとし、出力が１１ビットの非線形のテーブルを作成する。

【００４８】

以上によって、最適化後のＳ－ｂｏｘの組み合わせ数に対応する合成テーブル（非線形テーブル）を作成できたこととなる。

図１０は、本発明のＳ－ｂｏｘとＬの合成フローチャートを示す。

【００４９】

図１０において、Ｓ６１は、拡大されたＳ－ｂｏｘを入力する。

Ｓ６２は、線形変換Ｌを入力する。

Ｓ６３は、拡大Ｓ－ｂｏｘの出力の、Ｌによる線形変換の結果を格納する。これは、例えば右側に示すように、拡大Ｓ－ｂｏｘの出力を、線形変換Ｌで変換した後の結果を出力するように、右側のテーブルに格納し、当該線形変換Ｌの処理を拡大Ｓ－ｂｏｘのテーブルに取り込む。

【００５０】

以上によって、線形変換Ｌの処理が拡大Ｓ－ｂｏｘに取り込まれ、暗号化／復

号化時に当該線形変換Ｌの処理が不要となり、高速化を図ることが可能となる。

図１１は、本発明のＦ関数例（実装時）を示す。ここでは、図１０で合成した後の拡大Ｓ－ｂｏｘである$S_{11}$、$S_{10}$、$S_{11}$を図示のように配置および他の回路（ＸＯＲなど）を配置することにより、Ｆ関数を生成することが可能となる。そして、当該生成したＦ関数をもとに既述した図６のＦｅｉｓｔｅｌ構造データを作成し、平文を暗号化して暗号文を出力したり、暗号文を復号化して平文を出力したりすることが可能となる。この際、計算機が持つ一次キャッシュメモリ上にテーブルを配置して参照回数を最小限にして高速に暗号化／復号化の処理を実行することが可能となる。

【００５１】

【発明の効果】

以上説明したように、本発明によれば、計算機毎の一次キャッシュメモリ量および全体の入出力ビット数をもとにＳ－ｂｏｘの入出力数を最適化する構成を採用しているため、計算機毎に最適な高速な暗号化／復号化を行う装置を設計したり、実装したりすることが可能となる。

【図面の簡単な説明】

【図１】

本発明のシステム構成図である。

【図２】

本発明の全体動作説明フローチャートである。

【図３】

本発明の最適化手順フローチャートである。

【図４】

本発明のＳ－ｂｏｘの生成フローチャートである。

【図５】

本発明のＦ関数例（設計時）である。

【図６】

本発明のＦｅｉｓｔｌ構造データ例である。

【図７】

本発明の動作説明フローチャート（実装時）である。

【図８】

本発明のＳ－ｂｏｘの抽出フローチャートである。

【図９】

本発明の合成テーブルの作成フローチャートである。

【図１０】

本発明のＳ－ｂｏｘとＬの合成フローチャートである。

【図１１】

本発明のＦ関数例（実装時）である。

【符号の説明】

１：処理装置

２：入力手段

３：Ｓ－ｂｏｘの最適化手段

４：Ｓ－ｂｏｘの生成手段

５：Ｆ関数の生成手段

６：入力ファイル

７：出力ファイル

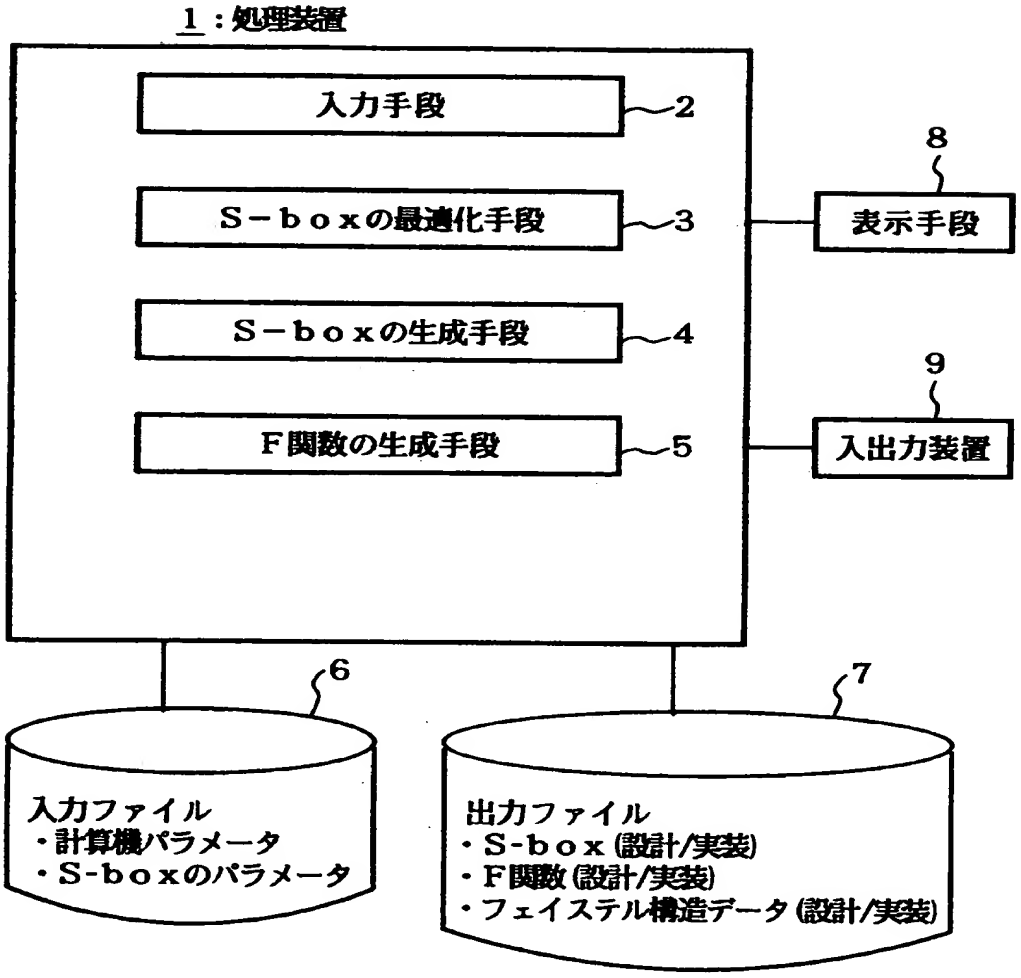
８：表示装置

９：入出力装置

【書類名】　　図面

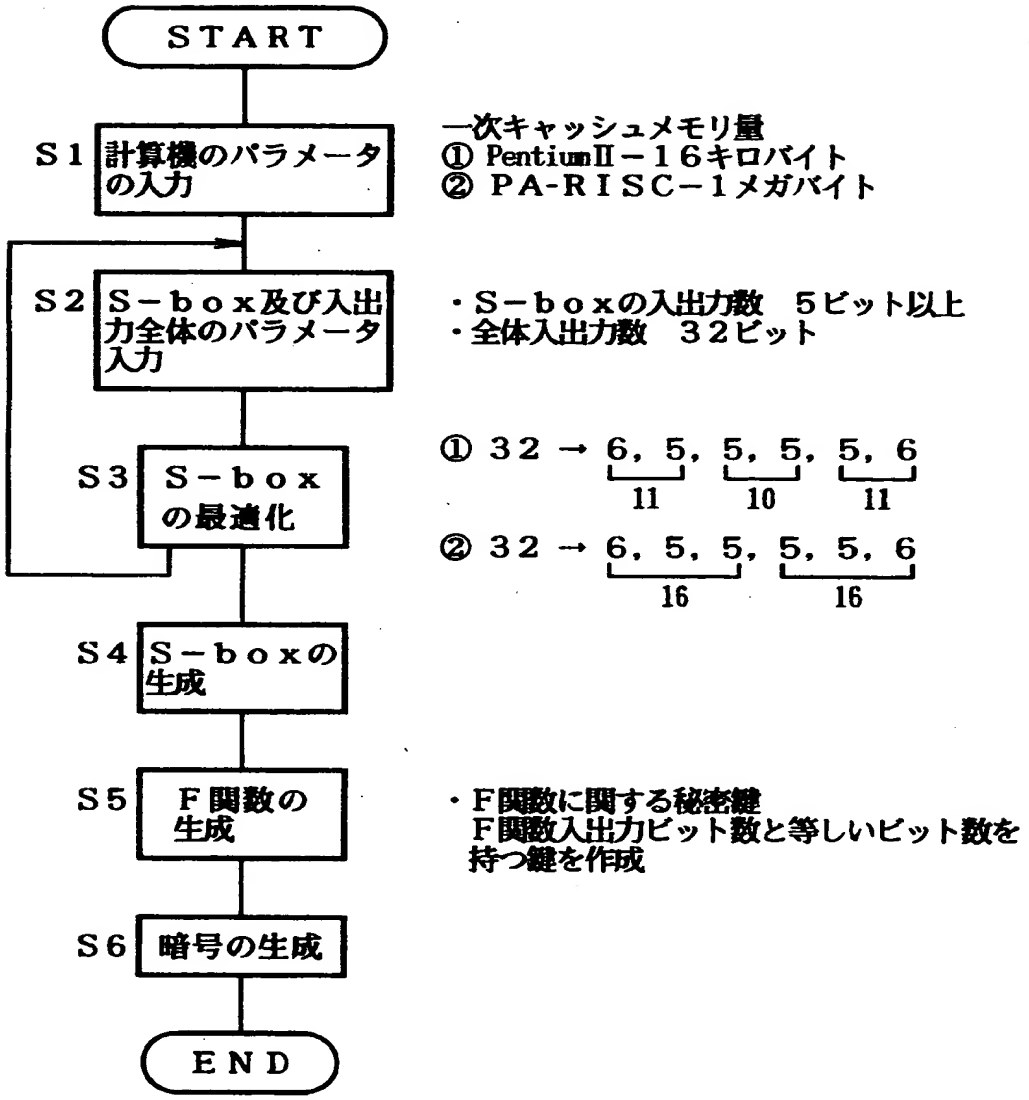
【図１】

本発明のシステム構成図



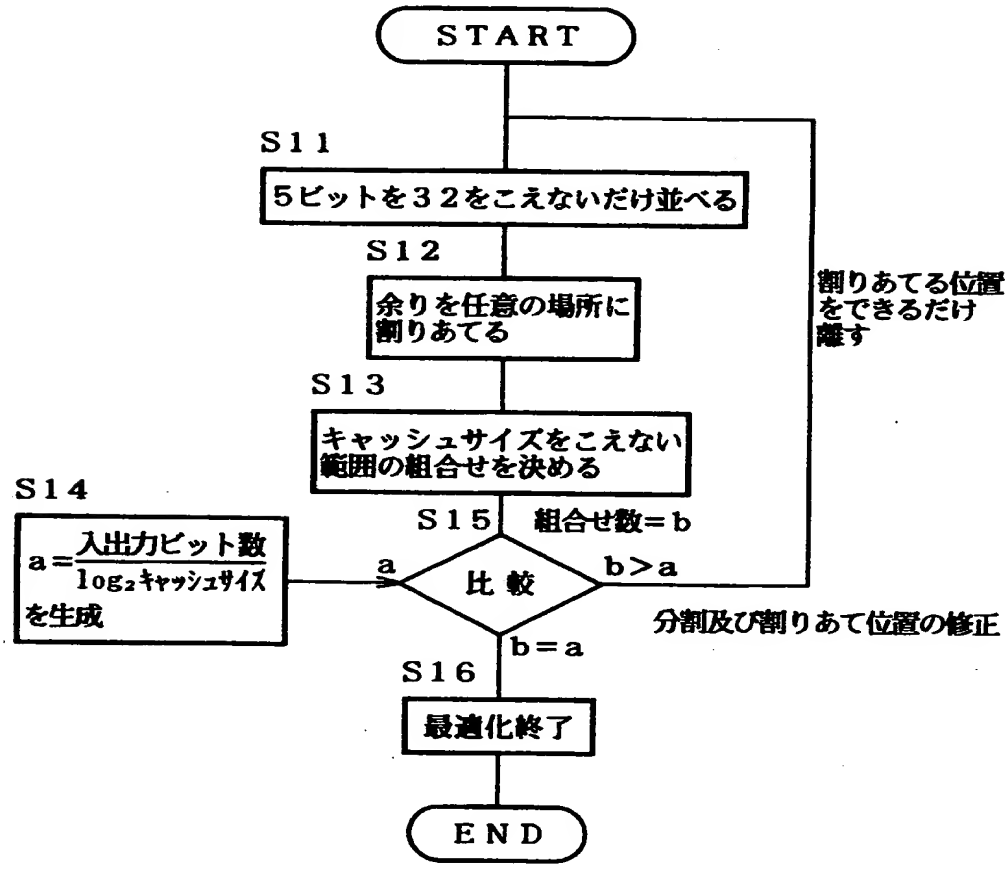
【図２】

本発明の全体動作説明フローチャート

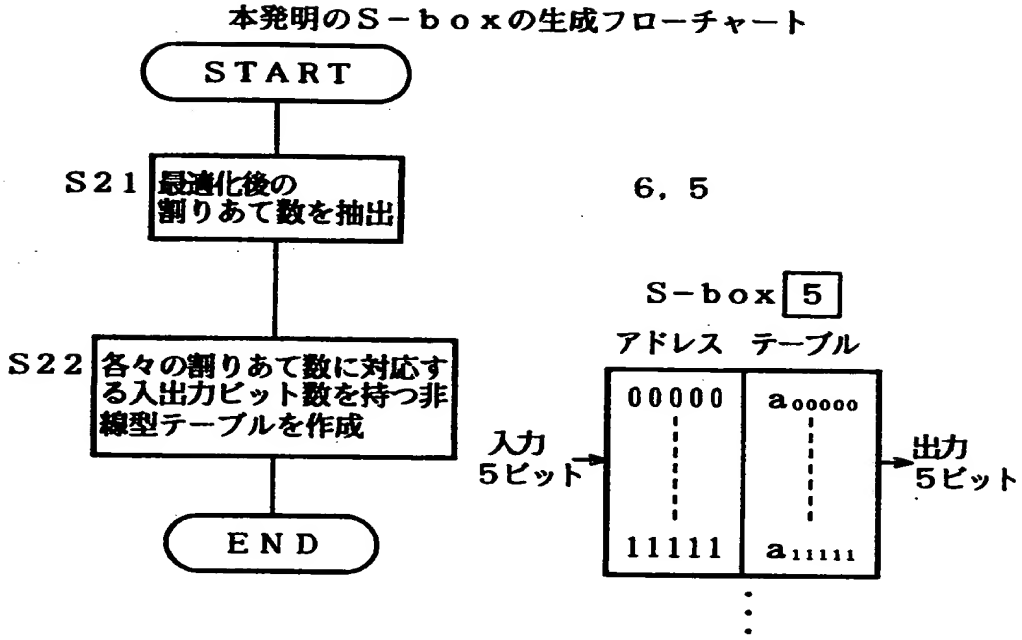


【図３】

本発明の最適化手順フローチャート

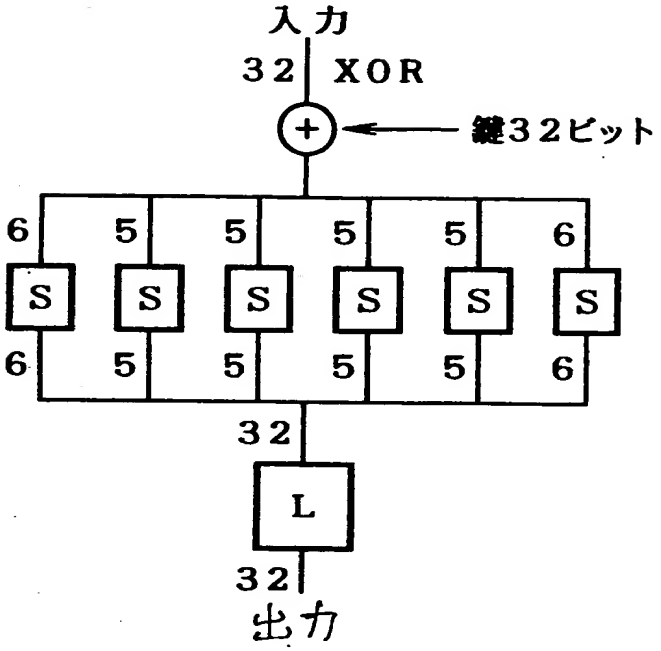


【図４】


本発明のＳ－ｂｏｘの生成フローチャート
START
S21
最適化後の
割りあて数を抽出
S22
各々の割りあて数に対応する入出力ビット数を持つ非線型テーブルを作成
END
6，5
S-box 5
アドレス テーブル
00000
11111
入力
5ビット
出力
5ビット

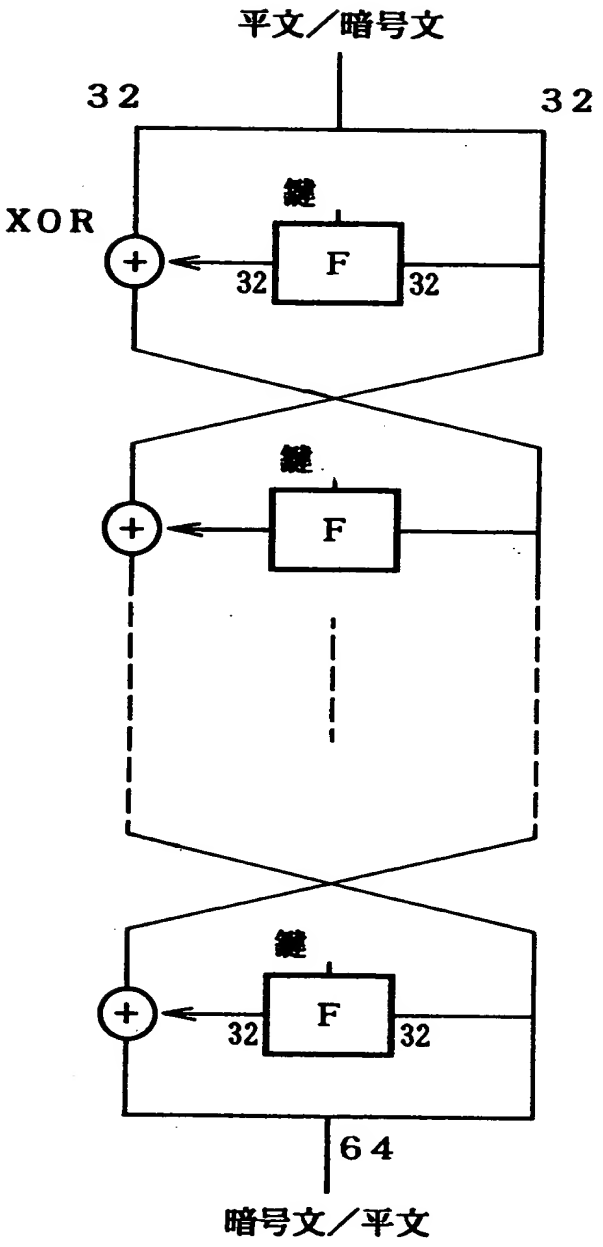
【図５】

本発明のＦ関数例（設計時）



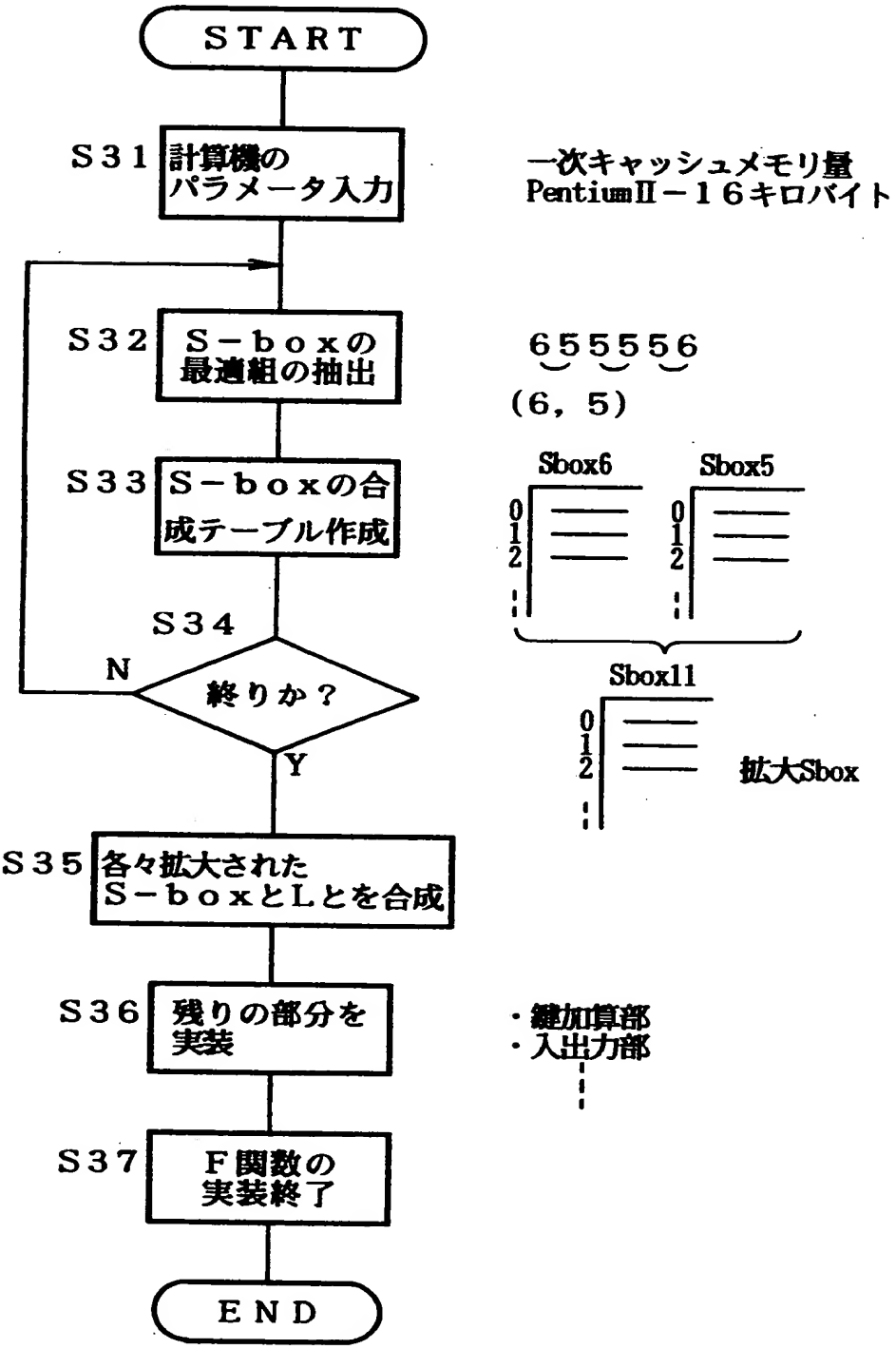
【図６】

本発明のＦｅｉｓｔｅｌ構造データ例



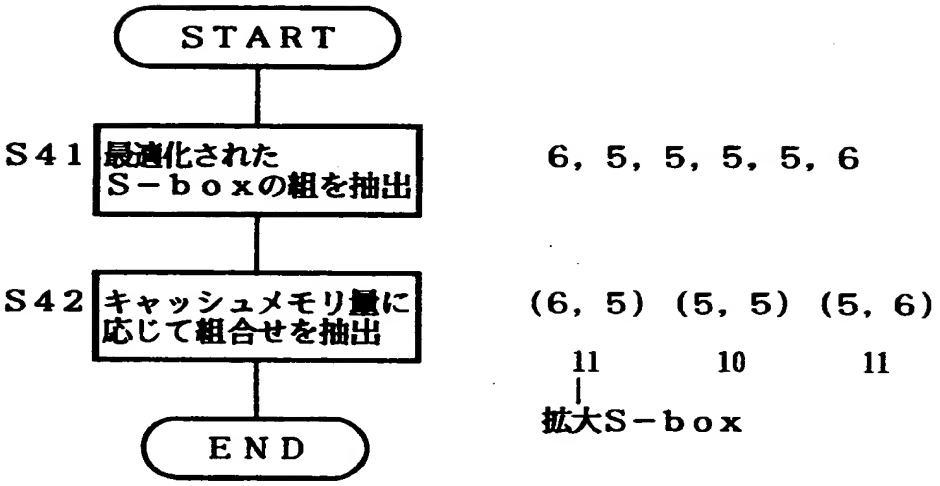
【図７】

本発明の動作説明フローチャート（実装時）



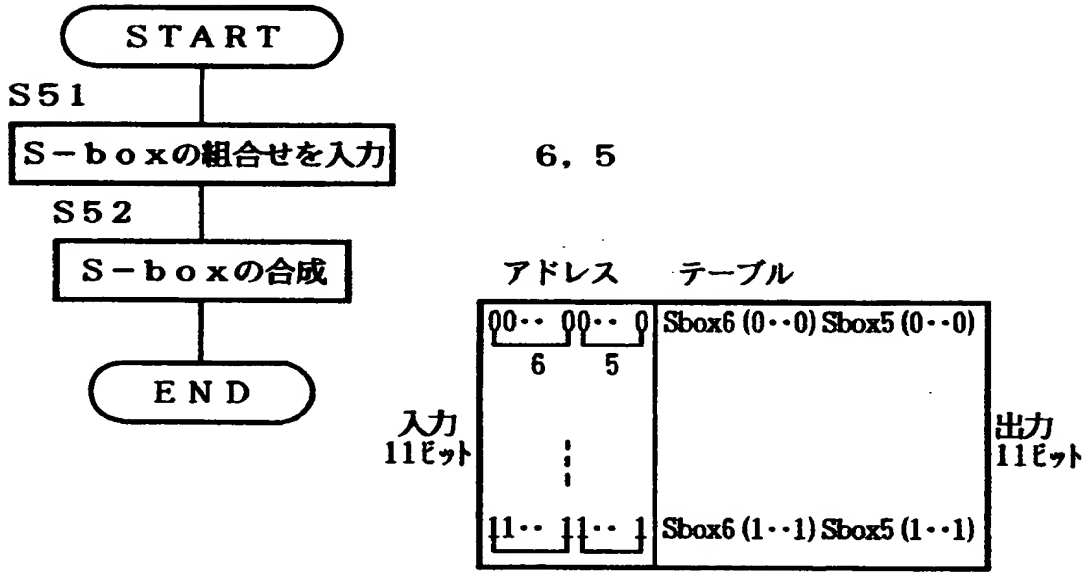
【図８】

本発明のＳ－ｂｏｘの抽出フローチャート



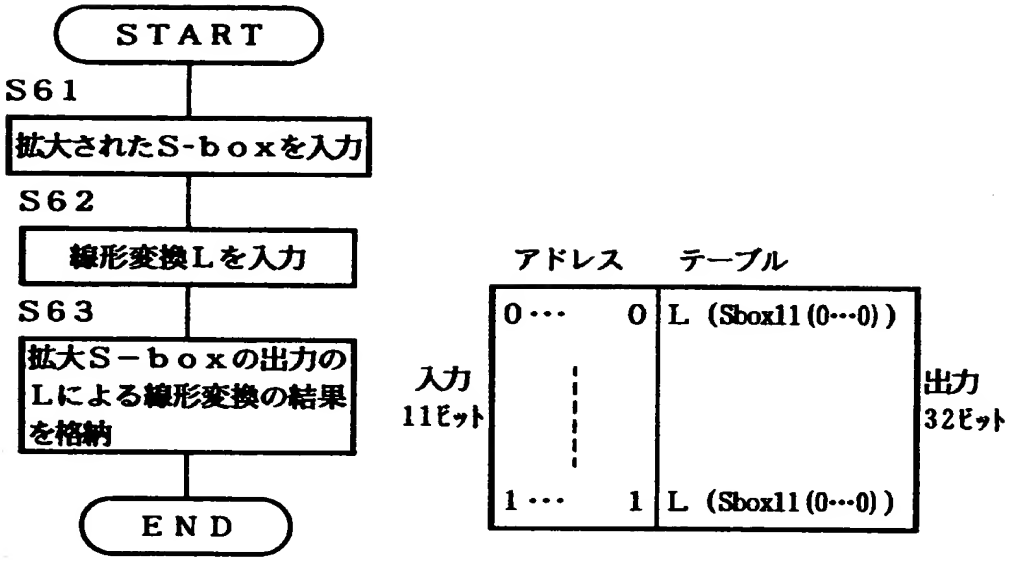

【図９】

本発明の合成テーブルの作成フローチャート



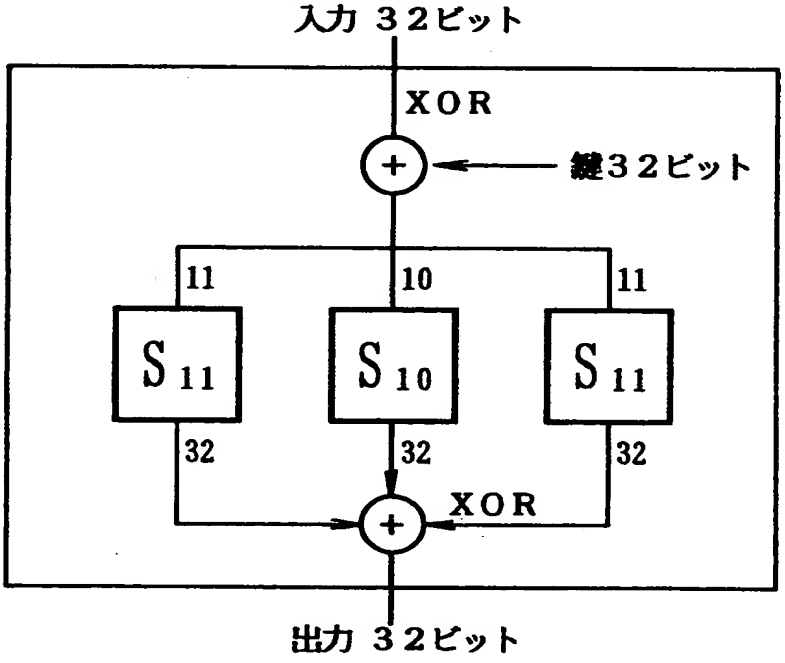
【図１０】

本発明のＳ－ｂｏｘとＬの合成フローチャート




【図１１】

本発明のＦ関数例（実装）



【書類名】　要約書

【要約】

【課題】　本発明は、計算機に最適な入出力数のＳ－ｂｏｘを持つ暗号化装置を設計する暗号設計装置および記録媒体に関し、計算機毎の一次キャッシュメモリ量および全体の入出力ビット数をもとにＳ－ｂｏｘの入出力数を最適化し、計算機毎に最適な高速な暗号化／復号化を実現することを目的とする。

【解決手段】　計算機の一次キャッシュメモリ量、および全体の入出力ビット数を入力する手段と、入力された全体の入出力ビット数を分割してＳ－ｂｏｘの入出力数を生成し、余りがでたときに任意のＳ－ｂｏｘの入出力数に割り当ててＳ－ｂｏｘの入出力数を仮決定する手段と、仮決定したＳ－ｂｏｘの入出力数を組み合わせて、入力された一次キャッシュサイズを越えない範囲で当該組み合わせを行う手段とから構成する。

【選択図】　図１

# 出　願　人　履　歴　情　報

識別番号　　〔０００００５２２３〕

１．変更年月日　　１９９６年　３月２６日
［変更理由］　住所変更
住　所　神奈川県川崎市中原区上小田中４丁目１番１号
氏　名　富士通株式会社